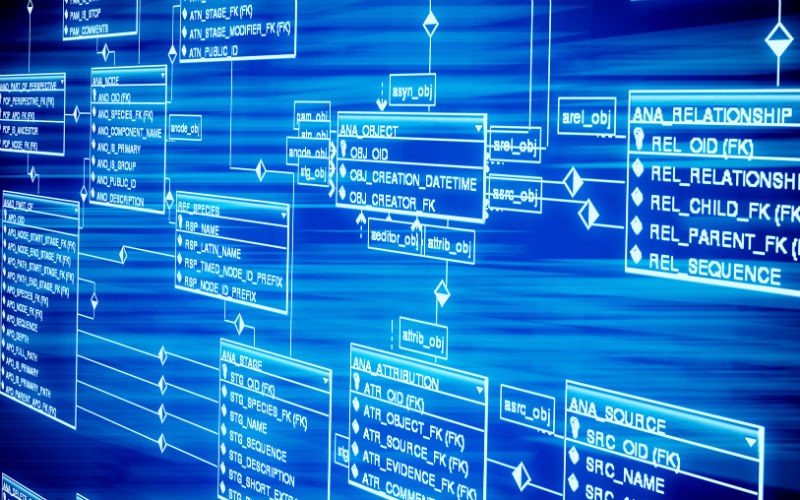


Engenharia Informática

Pedro Miguel da Costa Graça

David Sousa Pinto

**Bases de Dados**



**Índice**

1. Introdução 3
2. Introdução
3. Circuito Somador 4
4. Circuito Subtrator 7
5. Circuito Subtrator Complemento de 2 11
6. Circuito Comparador 15

**Introdução**

Para este trabalho foi escolhida como tema, uma API (Application programming interface) que fornece resultados de jogos em tempo real assim como informações de jogadores de diversos desportos.

Os objetivos desta base de dados serão armazenar e organizar corretamente os dados, como os tipos de desportos que a API possui, os países e as ligas que as constituem, assim como as equipas que lhes pertencem.

Apesar de não representar totalmente o nosso projeto conjunto que é, um site de resultados online que possui outras funcionalidades para além do uso desta API para fornecer os resultados, esta está ligada diretamente a ele, representando uma parte essencial do site, sem o qual este não seria possível.

Para a implementação correta da base de dados, de modo a ela ser 100% funcional e livre de erros, foram desenhados e projetados vários tipos de modelos com o propósito de verificar e testar o funcionamento da base de dados.

Modelo Entidade-Associação

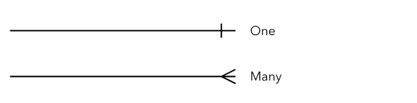
O primeiro modelo concebido foi o de entidade-associação para desenhar a estrutura da base de dados e organizar as ligações necessárias entre as entidades criadas.

Abaixo, uma imagem do modelo entidade-associação:

O esquema apresentado acima, não possui referências a chaves estrangeiras, estando estas representadas no modelo que se encontra abaixo, o modelo relacional.

Voltando agora ao modelo entidade-associação, este é composto por 9 entidades e, 2 associações com atributos próprios.

A multiplicidade é representada pelos símbolos:



É também utilizado o símbolo , para representar uma associação ISA.

Todas as entidades têm atribuída a multiplicidade correta, exceto as associações com atributos próprios (mais concretamente, “Participantes” e “Epoca”), estas apenas têm a ligação às entidades que ela interliga.

Para a construção deste modelo, consideramos 9 entidades como já referi, sendo estas:

- Equipa

- Desporto

- Jogo

- Resultado

- Hoquei

- Tenis

- Pais

- Jogador

- Liga

Consideramos também, 2 associações com atributos próprios:

- Epoca

- Participantes

Como a API apresenta resultados online como já referimos de diversos desportos, é essencial termos uma entidade (“Desporto”) que represente os vários tipos de desportos que a API suporta.

Como estamos a falar de desportos, é obrigatório termos entidades que representem, as ligas que esse desporto possui, o país a que essas ligas pertencem, e, as equipas que cada liga tem.

Para isto, foram criadas as entidades “Pais”, “Liga” e “Equipa”.

Após tudo isto, foi necessário criar uma entidade que representasse os jogos que irão ocorrer, mostrando informações sobre esse jogo como por exemplo, a hora a que ele se inicia e em que estádio irá decorrer a partida, e também, uma associação denominada “Participantes”, de modo a ser possível mostrar quais as equipas a defrontar-se nesses jogos.

Mas, tudo isto não fará sentido senão tivermos um método para representar o resultado de cada jogo, que é o aspeto mais importante.

Para resolver este problema, criou-se então uma entidade chamada “Resultado” que irá mostrar os golos que cada equipa marcou.

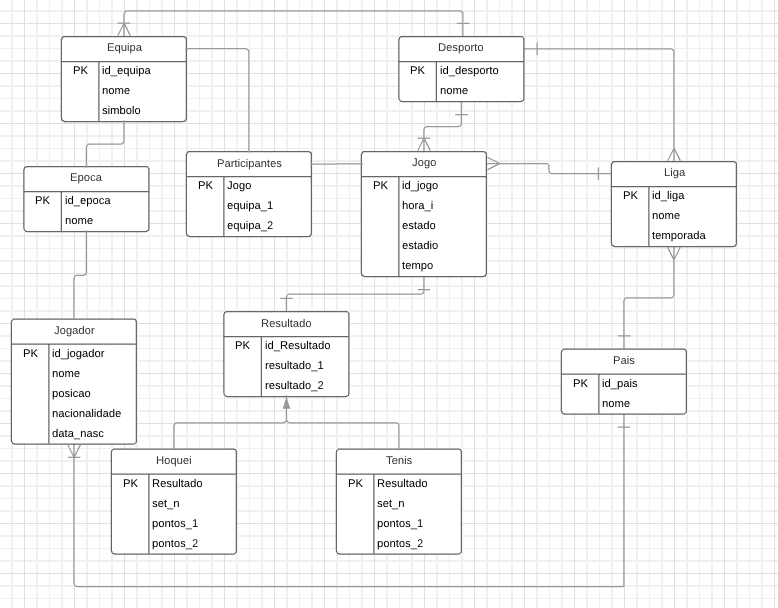
Como não existe só futebol, apesar de em Portugal esta ser a realidade, tivemos de criar mais 2 entidades que tornassem possível representar desportos, em que os resultados têm um formato mais complexo, como por exemplo, Ténis e Hóquei em que existem vários períodos durante um jogo, tendo cada período um número de pontos marcado por cada equipa.

Para os outros desportos, o modelo usado no futebol é compatível e pode ser usado para representar os resultados, não sendo necessário criar mais nenhuma entidade.

Como referimos antes, a API mostra também, para além de resultados online, informações de jogadores, como por exemplo em que equipa é que eles estão na época atual, assim como a sua data de nascimento e o seu país de origem. Foram então criadas, a entidade “Jogador” que possui as características pessoais do mesmo, e uma associação chamada “Epoca” onde serão guardadas a data da época, e o nome da equipa onde o jogador

esteve nessa época.

Modelo Relacional

Tendo representado o modelo entidade-associação, passamos agora ao modelo relacional:

No modelo relacional são agora referidas todas as ligações entre as entidades, mostrando as chaves estrangeiras e a qual atributo de uma certa entidade ela faz referência.

SQL como LDD

Após a criação e verificação dos modelos entidade-associação e modelo relacional, procedeu-se à implementação em SQL da base de dados.

Criação da base de dados e respetivas tabelas:

